

Arduino 应用入门

一、Arduino 简介



Arduino 是一款简单易用、便于上手的开源电子原型平台，包含硬件部分（各种符合 Arduino 规范的开发板）和软件部分（Arduino IDE 和相关的开发包）。

硬件部分（或称开发板）由微控制器（MCU）、闪存（Flash）以及一组通用输入/输出接口（GPIO）等构成，你可以将它理解为是一块微型电脑主板。

软件部分则主要由 PC 端的 Arduino IDE 以及相关的板级支持包（BSP）和丰富的第三方函数库组成。使用者可以借由 Arduino IDE 轻松地下载你所持有的开发板相关的 BSP 和需要的函数库，用于编写你的程序。

Arduino 是一个开源平台，至今已经有了多种型号及众多衍生控制器，包括 Arduino Uno、Arduino Nano、ArduinoYún 等等。另外，Arduino IDE 如今也不仅仅支持 Arduino 系列开发板，还以引入 BSP 的方式增加了对 Intel Galileo、NodeMCU 等流行开发板的支持。

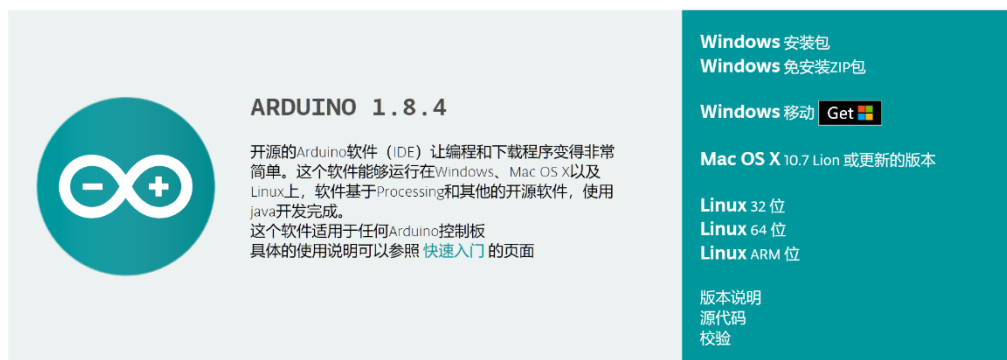
*注：我院物联网实验室（C102）主要提供 Arduino Uno 及 NodeMCU 两款实验平台

二、配置开发环境

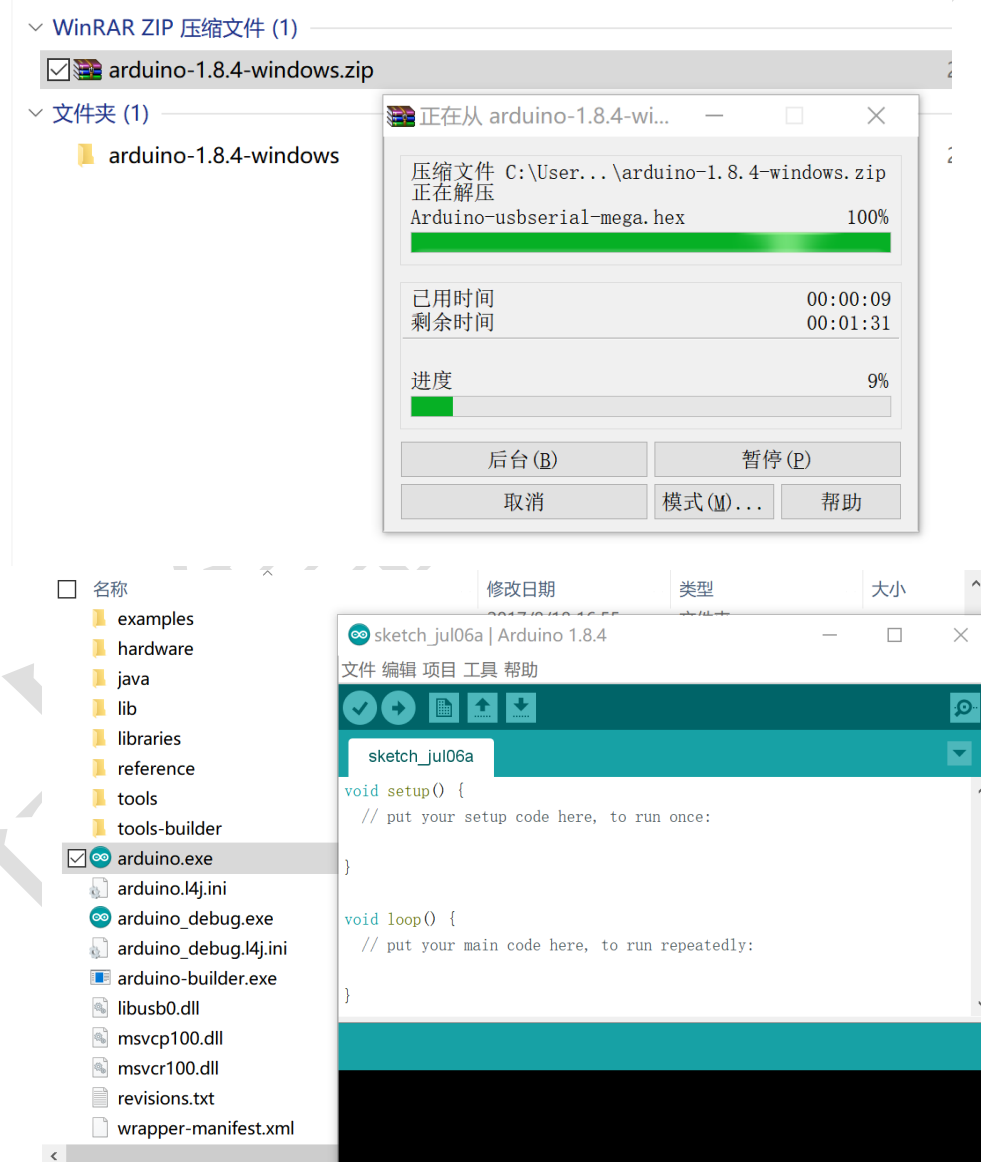
1. 通过下载安装包安装 Arduino IDE

登陆 Arduino 官方网站软件下载页 <https://www.arduino.cc/en/Main/Software?setlang=cn>

下载Arduino软件



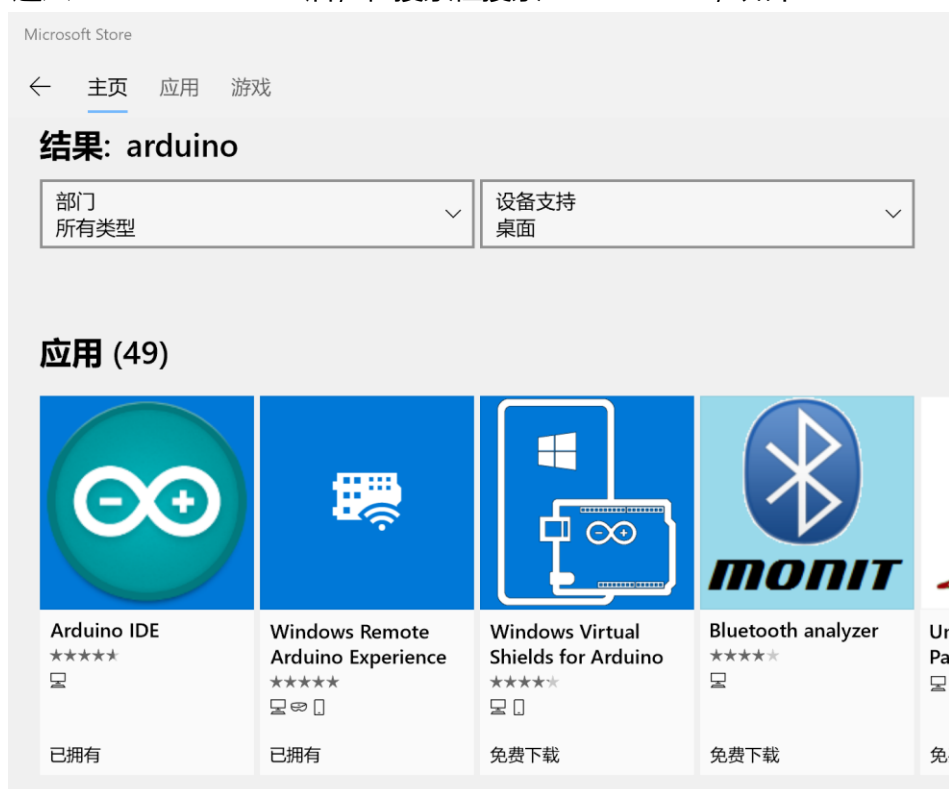
如果你的系统是 win7/8/10，推荐在浏览器内点击并下载“Windows 免安装 ZIP 包”，下载到你所指定的任意文件目录后解压



然后进入解压后的文件夹，点击“arduino.exe”，正常运行出现如图所示的程序窗口则环境配置成功

2. 通过 Windows 10 Store 安装 Arduino IDE

如果你的系统是 win10，也可以通过 Microsoft Store（在开始菜单中可以找到）安装 Arduino IDE，进入 Microsoft Store 后，在搜索栏搜索“Arduino”，如下



点选“Arduino IDE”，跳转到如下页面，点击“安装”即可，随后 Arduino IDE 的图表会出现在你的开始菜单当中



3. 其他平台

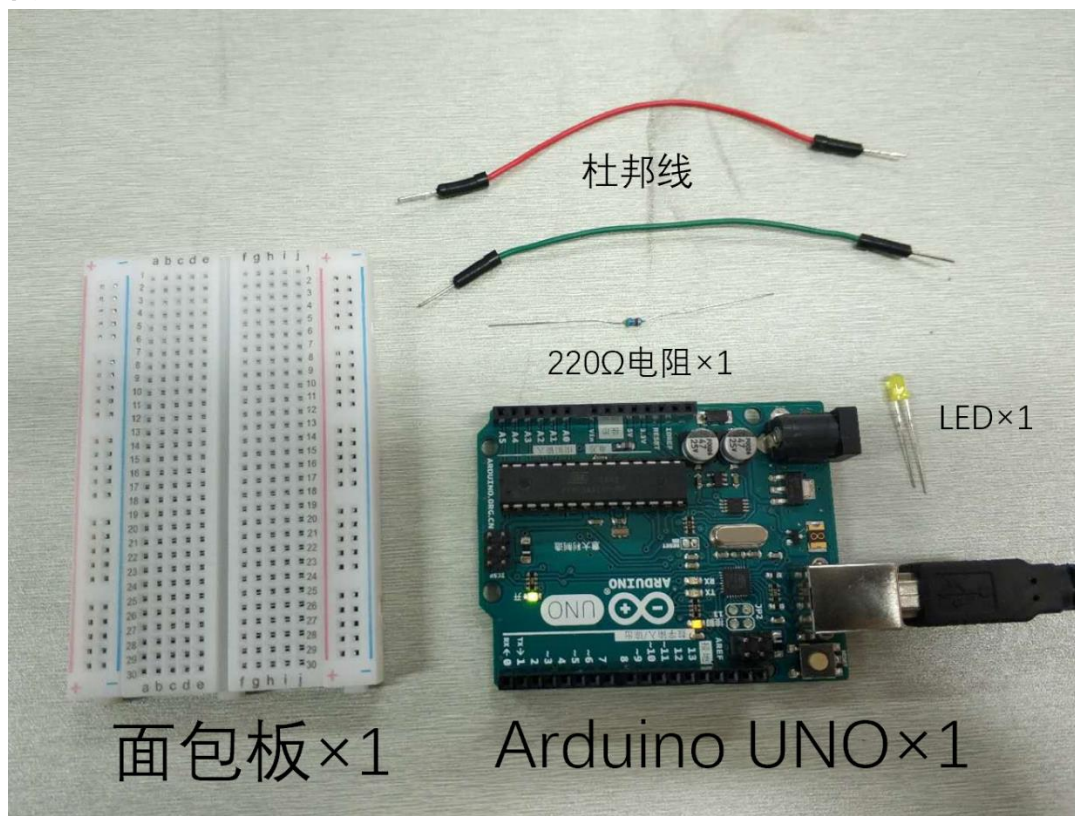
如果你使用 Linux 或 Mac OS X 等系统，请参考 Arduino 官网的安装示例配置你的环境

三、完成你的第一个应用

学习一门编程语言是从打印出第一行“hello world”开始，而学习嵌入式则是从点亮第一个 led 开始的，下面就以一个控制 led 的例子来让大家了解一个 Arduino 项目是如何工作的。

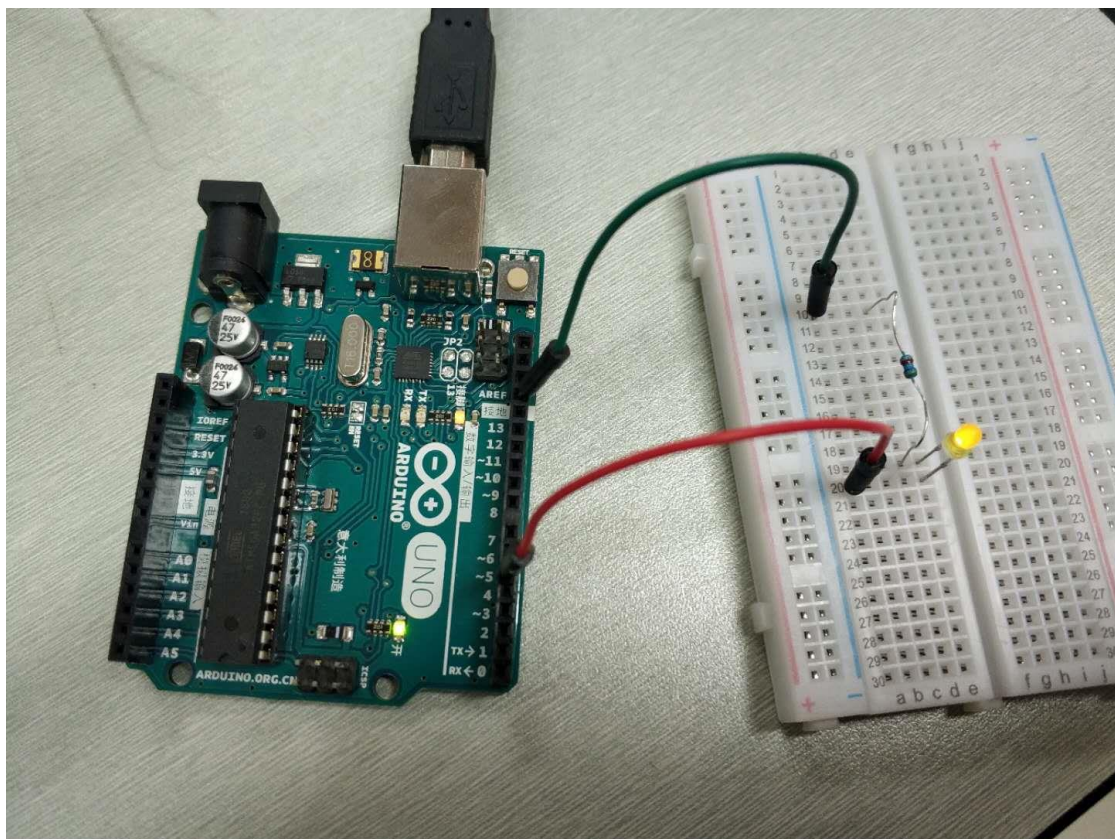
实验器材：Arduino UNO 开发板×1、USB 转接线（Micro USB Type-B）×1、面包板×1、LED 二极管×1、220Ω电阻×1、杜邦线×2

实验环境：Arduino IDE 1.8.4



*注：1) 杜邦线：即实验用导线；2) 面包板：用于无焊接实验的电路串并联接线，板上每一排 5 个插口（abcde）间相互导通，排与排之间相互独立；3) 色环电阻：可以根据色环读取阻值；4) led 二极管：两个不同长度的引脚，较短的引脚接地。

Step 1: 硬件接线



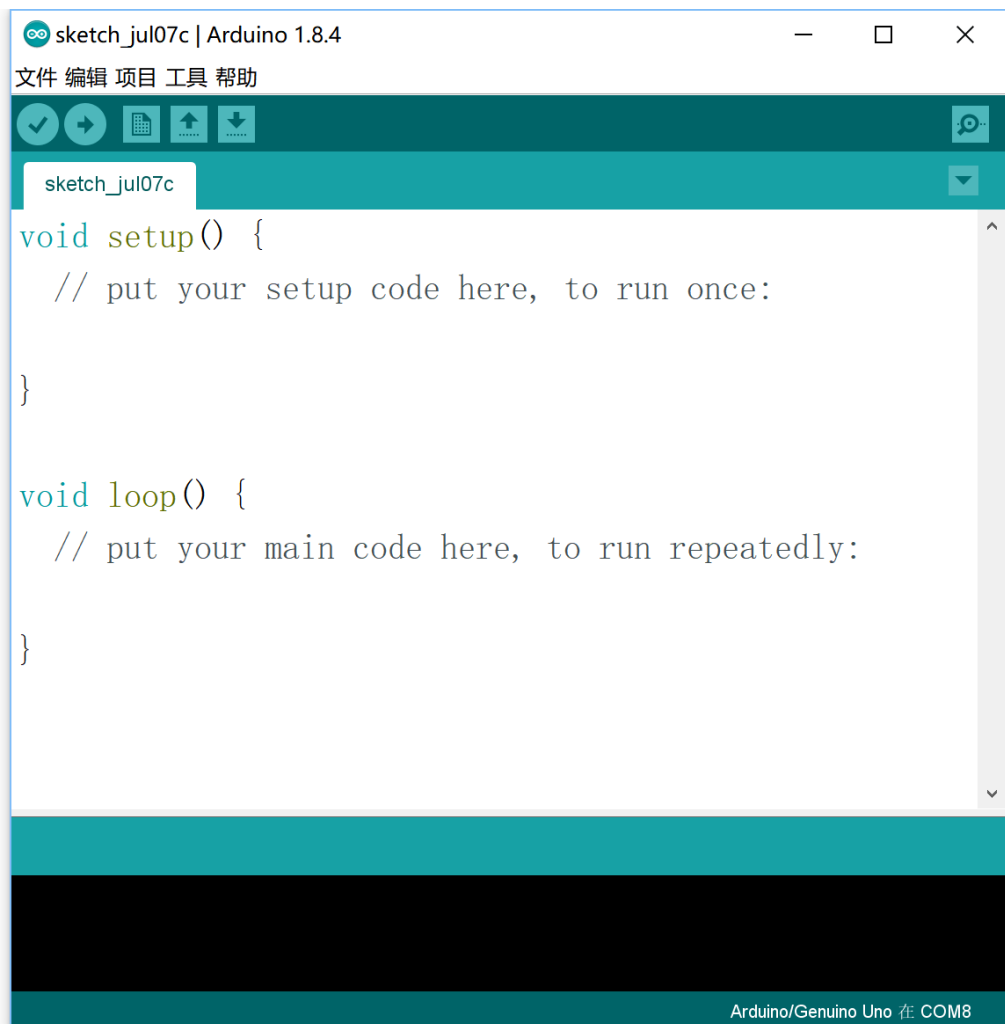
按照如图所示的方式连线：

- 1) USB 转接线一段连 PC 机的 USB 口，一端连到 Arduino UNO 的转接口上，用于从 PC 机上下载代码到开发板及供电；
- 2) 两条杜邦线分别一端接开发板的数字输入/输出 5 号（会在之后的代码中定义数字“5”）和接地两个接口，另一端接入面包板不同的两排（在本图中，若按照面包板上“字母/数字”的方式定位，绿色接入 a10，红色接入 a20，实际操作时可以接入任意独立的两排，不影响实验现象）；
- 3) 将二极管的长引脚插入面包板上红色杜邦线插入的排，短引脚插入相邻的排，然后将 220Ω 电阻的一端接短引脚，另一端接绿线（即接地），完成连线。

Step 2: 编写代码

1. 新建项目

初次打开 Arduino IDE 会有一个默认的新建项目，你也可以依次点击工具栏中“文件→新建”或使用 Ctrl+N 快捷键来创建一个新的项目，项目的命名和保存与大多数 IDE 和编辑器相同，这里就不做过多介绍。



2. 代码结构

Arduino IDE 项目使用简化版的 C/C++ 语言编写代码，只要你有一点点 C 语言学习的基础就很容易入门，新建项目默认有两个函数体：**void setup()** 和 **void loop()**，这两个函数体是一个 Arduino 项目所必须的。

同 C/C++ 代码一样，在代码的头部定义函数库的引用及全局变量等；

void setup() 是设备上电，即初始化会调用的函数，只执行一次，函数体内部一般包含引脚的定义信息；

void loop() 相当于 C 语言中的主函数，函数体内部的代码会在设备供电期间循环执行，你希望 Arduino 能完成的主要工作都写在这个函数体当中；

*注：除上述两个必要函数外，你也可以定义其他函数再在 void loop() 中调用，或将函数写在外部的 .cpp 文件中，再以像 C/C++ 项目的方式来调用

3. 编写代码

```
example $
/* 这里根据引脚接线定义变量ledPin为数字“5”
 * 你也可以改变接线位，同时修改这里的变量
 */
int ledPin = 5;

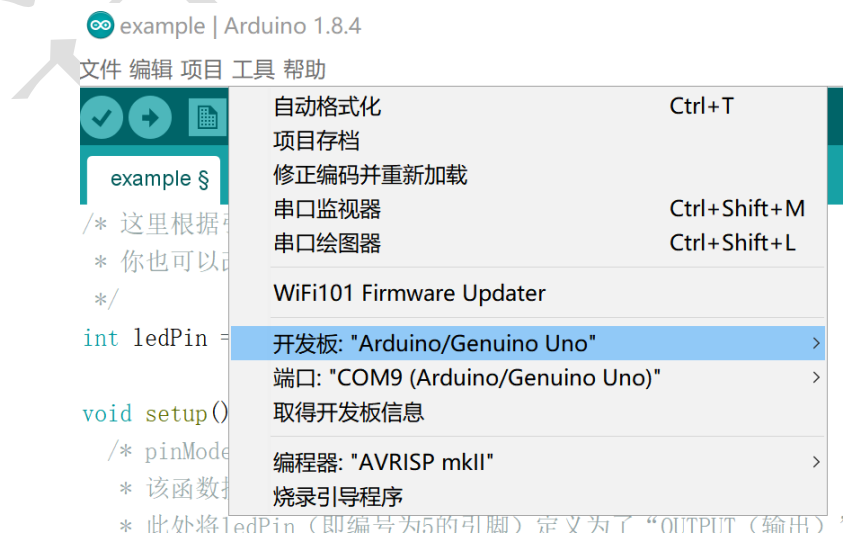
void setup() {
  /* pinMode()是Arduino项目的一个内置函数，用于引脚功能定义
   * 该函数接收两个参数，分别是引脚编号和输入/输出格式
   * 此处将ledPin（即编号为5的引脚）定义为了“OUTPUT（输出）”
   * 然后设备初始化的时候调用此函数，5号引脚便能一“输出”方式工作
   */
  pinMode(ledPin, OUTPUT);
}

void loop() {
  /* digitalWrite()用于向已被pinMode()设置为“输出”的引脚写入数字信号
   * 这里“HIGH”表示高电压（对于Arduino UNO来说是5V），“LOW”表示低电压（0V）
   * delay()接收一个整数参数，表示暂停相应时间（单位为毫秒）
   */
  digitalWrite(ledPin, HIGH); // 向ledPin写入高电压，led灯亮
  delay(1000); // 持续1000毫秒（1秒）
  digitalWrite(ledPin, LOW); // 向ledPin写入低电压，led灯灭
  delay(1000); // 持续1000毫秒（1秒）
}
```

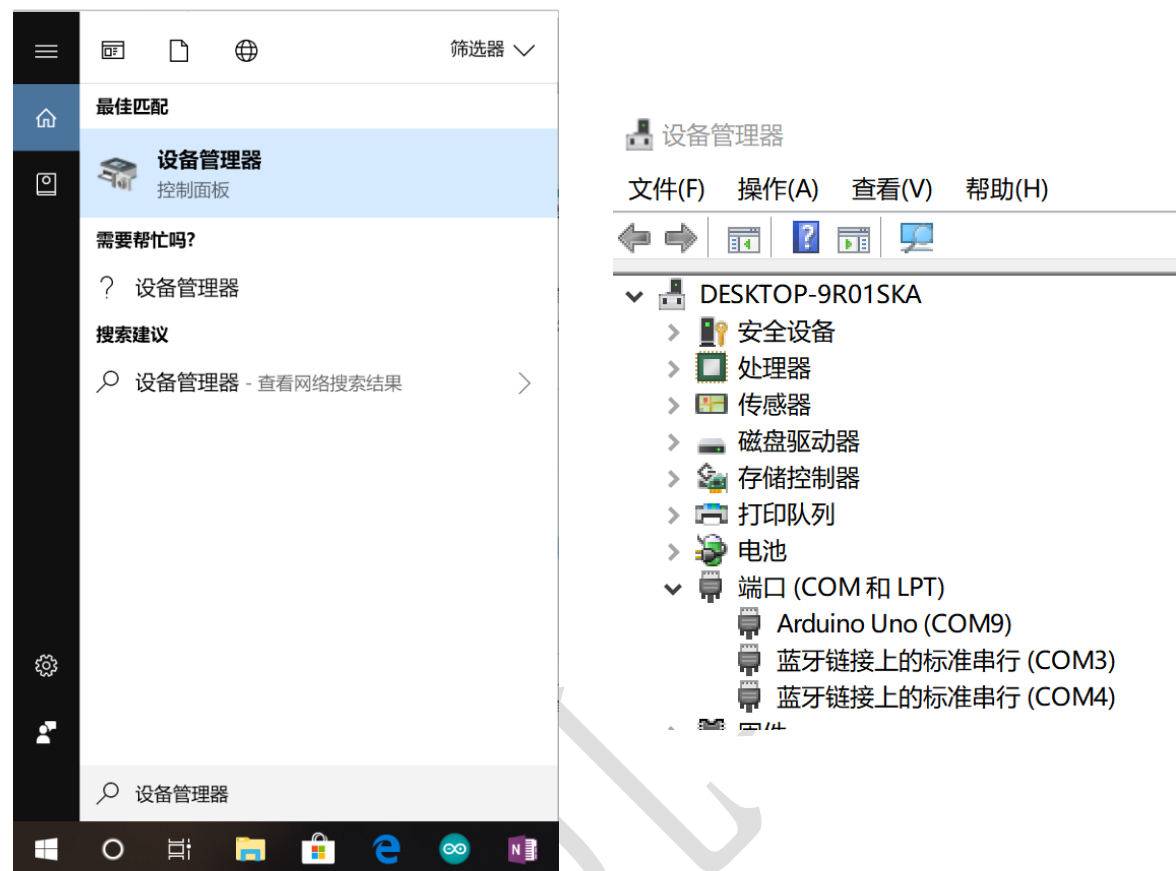
本例代码如上图，除去注释，控制一个灯的代码仅有短短数行

Step 3: 上传到开发板

编写好代码后，就要把代码通过数据线上传到开发板了，首先要确保编译条件匹配你的开发板并保持端口畅通，点击菜单栏“工具”，确保你的开发板选择为“Arduino/Genuino Uno”



端口号则因不同 PC 分配的端口号不同而有差异，笔者电脑上为“COM9”，这个信息可以在 windows 设备管理器中查看（可以通过 windows 开始菜单 搜索“设备管理器”，或在控制面板中打开）。



保证上述两个条件无误后，可以先点击菜单栏下 圆形√按钮 进行预编译/验证，如有问题，下方的信息栏会报错。

然后你可以直接点击菜单栏下的 圆形→按钮 或者点击“项目→上传”或使用快捷键 Ctrl+U 进行上传，等待片刻，IDE 会提示“上传成功”，然后就能观察到你刚刚连接的 led 灯不断闪烁的实验现象。

*注：上传前会重新编译，如果有问题也会报错，错误信息输出在下方的信息栏中